Mobile Price Predictor

-By Gaurvi Bhardwaj

Step 1: Prototype Selection

Abstract

The report aims to apply machine learning techniques to predict mobile phone prices based on their features. By analyzing various attributes such as battery life, RAM, internal storage, and more, we can develop a predictive model that can assist consumers in making informed purchasing decisions and manufacturers in pricing their products competitively.

1.Problem Statement

The mobile phone market is highly competitive, with numerous models being released every year. Predicting the price of a mobile phone based on its features can help both consumers and manufacturers. Consumers can find the best value for their money, while manufacturers can optimize their pricing strategies.

2.Market/Customer/Business Need Assessment

With rapid advancements in technology, consumers are overwhelmed with choices, and the price often becomes a decisive factor. Accurate price predictions based on features can simplify the decision-making process for consumers and provide manufacturers with data-driven insights to adjust their pricing.

3.Target Specifications and Characterization

- Prediction Accuracy: High accuracy in predicting prices to provide reliable information to users.

- Feature Analysis: Identify key features that most significantly impact mobile prices.
- Scalability: Ability to handle data from a large number of mobile phone models.

4.External Search (Information and Data Analysis)

These are some of the sources I visited for more information.

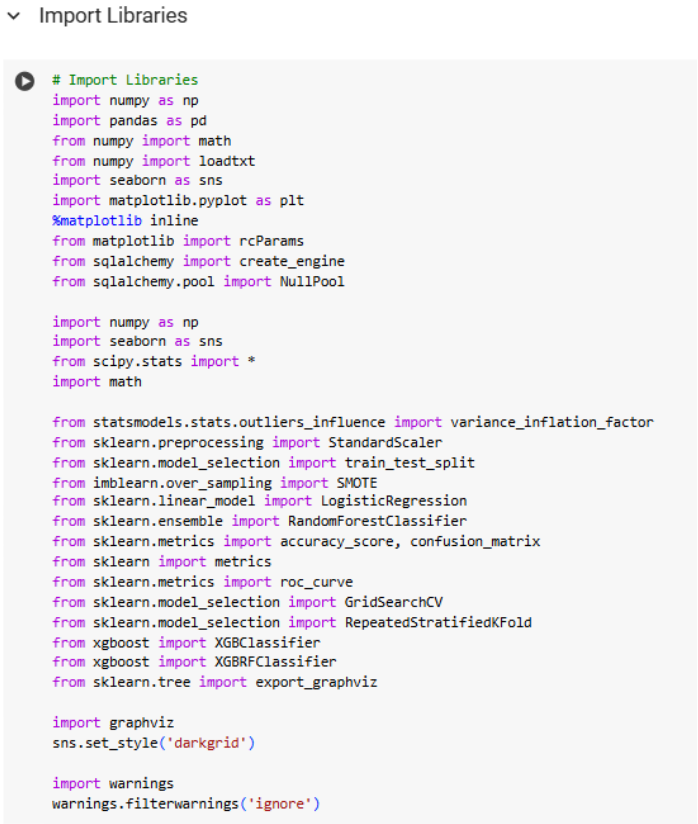
- GSMArena: Comprehensive database of mobile phone specifications and prices.
- Statista: Market analysis and statistics on mobile phone sales and trends.
- Kaggle: Datasets and competitions related to mobile phone price prediction.

using this Dataset for my code implementation for this report.

Dataset Description:

This is a data set related to all types of features of mobile phones which contains 2000 rows and 21 columns. In which there are no duplicate values and no missing values.

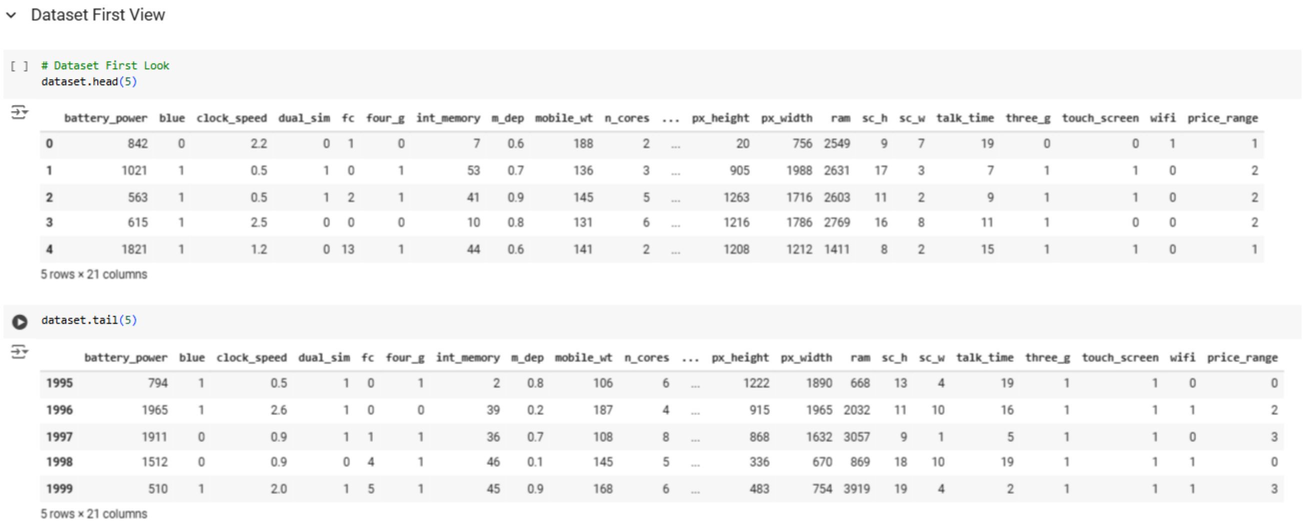
First import the basic libraries for data preprocessing:



Import Libraries

```
# Import Libraries
import numpy as np
import pandas as pd
from numpy import math
from numpy import loadtxt
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
from matplotlib import rcParams
from sqlalchemy import create_engine
from sqlalchemy.pool import NullPool

import numpy as np
import seaborn as sns
from scipy.stats import *
import math

from statsmodels.stats.outliers_influence import variance_inflation_factor
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from imblearn.over_sampling import SMOTE
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, confusion_matrix
from sklearn import metrics
from sklearn.metrics import roc_curve
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import RepeatedStratifiedKFold
from xgboost import XGBClassifier
from xgboost import XGBRFClassifier
from sklearn.tree import export_graphviz

import graphviz
sns.set_style('darkgrid')

import warnings
warnings.filterwarnings('ignore')
```

Let’s see more info on dataset:



Dataset First View

```
# Dataset First Look
dataset.head(5)
```

	battery_power	blue	clock_speed	dual_sim	fc	four_g	int_memory	m_dep	mobile_wt	n_cores	...	px_height	px_width	ram	sc_h	sc_w	talk_time	three_g	touch_screen	wifi	price_range
0	842	0	2.2	0	1	0	7	0.6	188	2	...	20	756	2549	9	7	19	0	0	1	1
1	1021	1	0.5	1	0	1	53	0.7	136	3	...	905	1988	2631	17	3	7	1	1	0	2
2	563	1	0.5	1	2	1	41	0.9	145	5	...	1263	1716	2603	11	2	9	1	1	0	2
3	615	1	2.5	0	0	0	10	0.8	131	6	...	1216	1786	2769	16	8	11	1	0	0	2
4	1821	1	1.2	0	13	1	44	0.6	141	2	...	1208	1212	1411	8	2	15	1	1	0	1

5 rows × 21 columns

```
dataset.tail(5)
```

	battery_power	blue	clock_speed	dual_sim	fc	four_g	int_memory	m_dep	mobile_wt	n_cores	...	px_height	px_width	ram	sc_h	sc_w	talk_time	three_g	touch_screen	wifi	price_range
1995	794	1	0.5	1	0	1	2	0.8	106	6	...	1222	1890	668	13	4	19	1	1	0	0
1996	1965	1	2.6	1	0	0	39	0.2	187	4	...	915	1965	2032	11	10	16	1	1	1	2
1997	1911	0	0.9	1	1	1	36	0.7	108	8	...	868	1632	3057	9	1	5	1	1	0	3
1998	1512	0	0.9	0	4	1	46	0.1	145	5	...	336	670	869	18	10	19	1	1	1	0
1999	510	1	2.0	1	5	1	45	0.9	168	6	...	483	754	3919	19	4	2	1	1	1	3

5 rows × 21 columns

5.Benchmarking

Benchmarking involves comparing project processes and performance metrics to either industry best standards and practices or successful completed projects. For this there is a need to continuously search for implementation of better techniques which lead to better results or outputs.

6.Applicable Regulations(Government and Environmental)

- Data collection and Privacy of Regulations of Customers.
- Data Privacy Regulations: Ensuring compliance with data protection laws (e.g., GDPR).
- Consumer Protection Laws: Accurate and fair pricing information to prevent deceptive practices.
- Rules against False Marketing
- Employment Schemes and laws created by government

7.Applicable Constraints

- Lack of initial data to perform algorithms.
- Data Quality: Ensuring the dataset is comprehensive and accurate.
- Feature Variability: Handling a wide range of features with varying degrees of impact on price.
- Market Changes: Adapting to rapid changes in technology and market trends.

9.Business Opportunity

Implementing machine learning for price prediction and customer analysis can significantly benefit small-scale mobile retailers.

- Machine learning models can analyze vast amounts of data to identify trends and patterns that might not be visible through manual analysis. This helps retailers make informed decisions about pricing, promotions, and product offerings (Kaggle) .
- By understanding customer preferences and purchasing behavior, retailers can tailor their product offerings to meet customer needs better. For instance, identifying which mobile features are most popular can guide stock and marketing strategies .
- Accurate price prediction models can forecast demand for different mobile models, helping retailers manage their inventory more efficiently. This reduces the risk of overstocking or stockouts, ensuring that the right products are available at the right time .

- Analyzing shopping baskets helps understand which mobile accessories or additional services (like warranties) are frequently bought together with certain phone models. This can lead to effective cross-selling and up-selling strategies .
- Personalized Marketing: Retailers can create targeted marketing campaigns based on customers' purchase history and preferences, increasing the likelihood of repeat purchases and customer loyalty.
- Personalized Recommendations: Machine learning can provide personalized recommendations to customers, enhancing their shopping experience and increasing satisfaction .
- Customer Retention: By understanding and predicting customer needs and preferences, retailers can improve customer retention through personalized services and targeted loyalty programs .

10.Concept Generation

Goal: Develop initial ideas and concepts for the mobile price prediction application.

Steps:

Identify User Needs: Understanding the target users and their needs. In this case, users might include mobile phone retailers, consumers, and tech enthusiasts who want to estimate the price of a mobile phone based on its features.

Brainstorming: potential concepts might include:

- A simple input form where users enter mobile phone specifications.
- Integration with a database of existing mobile phone prices for reference.
- A recommendation system suggesting similar phones within the user's budget.
- Visualization tools to compare features vs. price.

Market Research: Analyzing existing solutions and identify gaps. Also Looking at current mobile price prediction tools, their features, strengths, and weaknesses.

11.Concept Development

Goal: Refine the initial ideas into a more detailed and structured concept.

Steps:

Feature Specification: Determine the key features required for the application. Based on the screenshot, features might include:

Input fields for various mobile specifications (battery, Bluetooth, processor speed, dual SIM, camera, 4G support, internal memory, depth, weight, processor cores).

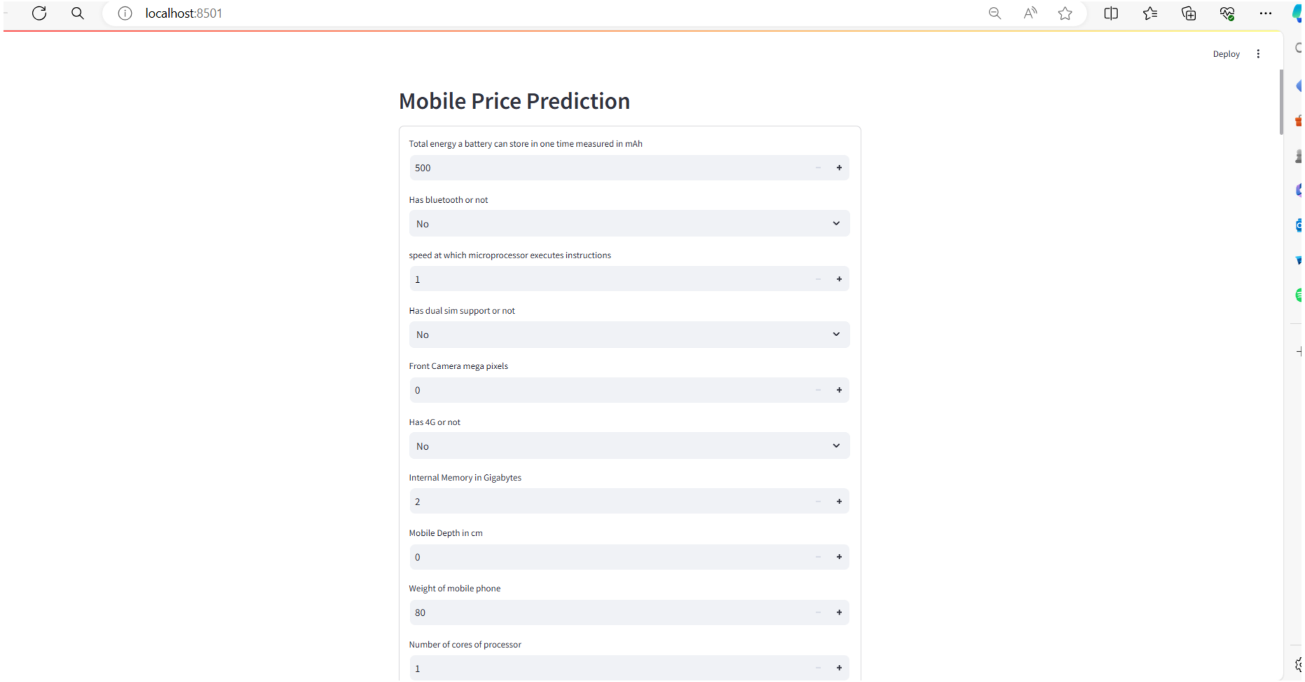
A prediction algorithm to estimate the price.

User-friendly interface for easy data entry and result viewing.

- Technical Feasibility: Assess the technical requirements and feasibility. This includes:
- Selecting the prediction model (e.g., machine learning algorithms like linear regression, decision trees, or neural networks).

 the technology stack (e.g., Python for the backend, Streamlit for the front-end) is used.

Final Results:



Mobile Price Prediction

Total energy a battery can store in one time measured in mAh

500

Has bluetooth or not

No

speed at which microprocessor executes instructions

1

Has dual sim support or not

No

Front Camera mega pixels

0

Has 4G or not

No

Internal Memory in Gigabytes

2

Mobile Depth in cm

0

Weight of mobile phone

80

Number of cores of processor

1

12.Final Product Prototype/ Product Details

The project is configured the development environment with necessary tools and libraries (e.g., Python, Streamlit, scikit-learn).

Backend Development:

Developed the machine learning model for price prediction. Training the model using historical data of mobile phone prices and their specifications.

Creating APIs for data processing and prediction.

Frontend Development:

Using Streamlit to create the user interface as seen in the screenshot. This includes developing forms for user input and displaying the predicted price.Ensure the interface is responsive and user-friendly.

the frontend with the backend enable end-to-end functionality, ensure that data flows correctly from user input to prediction output.

Performing thorough testing to identify and fix any bugs or issues. This includes unit testing, integration testing, and user acceptance testing.

A) Feasibility

The project is feasible with current technology and can be developed within a year. Data collection and model training will require significant initial effort.

B) Viability

As the mobile phone market continues to grow, the demand for such predictive tools will remain strong. Continuous updates and improvements will ensure long-term viability.

C) Monetization

The service can be monetized through subscription models, data licensing to manufacturers and retailers, and advertising within consumer applications.

Step 2: Prototype Development

Github: https://github.com/Gaurvi-bhardwaj/project3_mobile_price_prediction

Step 3: Business Modeling

- Subscription Model: Offer basic features for free with premium features available through a subscription.
- Data Analytics Services: Provide insights and analytics to mobile vendors for a fee.
- Advertising: Use customer data to offer targeted advertising for mobile accessories and related products.

Step 4: Financial Modeling

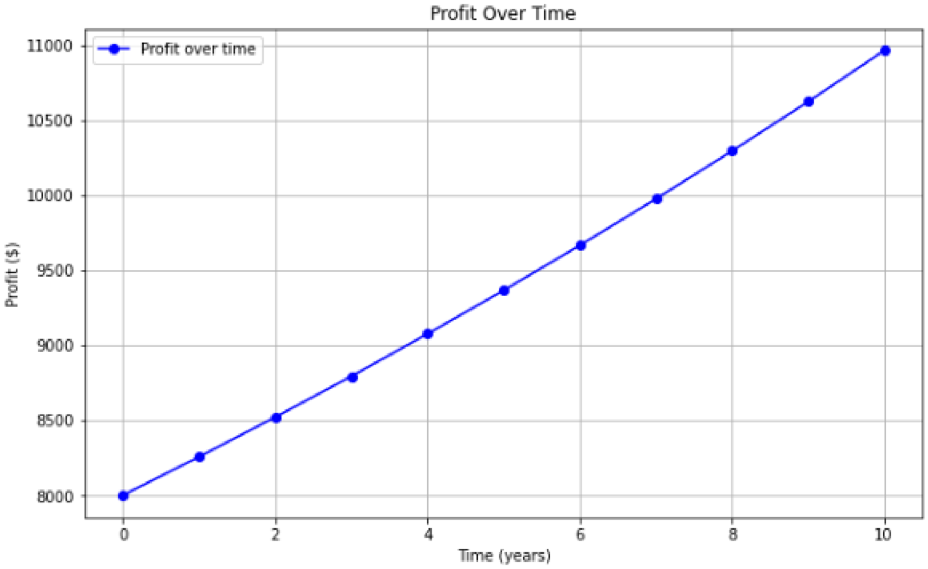
It can be directly launched into the retail market.

Let's consider price of product = 8000 for getting our graph

9

```python
import numpy as np
import matplotlib.pyplot as plt

initial_price = 8000 # Initial price of the product
growth_rate = 0.032  # Growth rate (3.2%)
time_period = 10  # Time period in years

# Time intervals (years)
t = np.arange(0, time_period + 1, 1)

# Calculate profit over time using the financial equation
profit = initial_price * (1 + growth_rate) ** t

# Print the profit values for each year
for year, profit_value in zip(t, profit):
    print(f"Year {year}: ${profit_value:.2f}")

# Plot the profit over time
plt.figure(figsize=(10, 6))
plt.plot(t, profit, marker='o', linestyle='-', color='b', label='Profit over time')
plt.title('Profit Over Time')
plt.xlabel('Time (years)')
plt.ylabel('Profit ($)')
plt.grid(True)
plt.legend()
plt.show()
```

```
Year 0: $8000.00
Year 1: $8256.00
Year 2: $8520.19
Year 3: $8792.84
Year 4: $9074.21
Year 5: $9364.58
Year 6: $9664.25
Year 7: $9973.51
Year 8: $10292.66
Year 9: $10622.02
Year 10: $10961.93
```



Conclusion

By implementing mobile price predictor, small-scale mobile retailers can gain a competitive edge in the market. These data-driven approaches enable better pricing strategies, improved inventory management, and enhanced customer satisfaction.